

[首页](#)[官方教程](#)[视频课程](#)[社区开源](#)[正在开发中.....](#)[登录](#)

## Spring Data JPA系列：分页（Pageable）

在JPA中提供了很方便的分页功能，那就是

Pageable (org.springframework.data.domain.Pageable) 以及它的实现类

PageRequest (org.springframework.data.domain.PageRequest)，详细的可以见示例代码：

### 1、改变CustomerRepository方法

```
/**
 * 一个参数，匹配两个字段
 * @param name2
 * @Param pageable 分页参数
 * @return
 * 这里Param的值和=:后面的参数匹配，但不需要和方法名对应的参数值对应
 * 这里增加了@QueryHints注解，是给查询添加一些额外的提示
 * 比如当前的name值为HINT_COMMENT是在查询的时候带上一些备注信息
 */
@QueryHints(value = { @QueryHint(name = HINT_COMMENT, value =
@Query("select c from Customer c where c.firstName=:name or c
Page<Customer> findByName(@Param("name") String name2,P
```

### 2、增加CustomerController方法pageable

```
/**
 * 分页
 * 应用查询提示@QueryHints，这里是在查询的适合增加了一个comment
 * 查询结果是lastName和firstName都是bauer这个值的数据
 */
@RequestMapping("/pageable")
public void pageable(){
    //Pageable是接口，PageRequest是接口实现
    //PageRequest的对象构造函数有多个，page是页数，初始值是0，size是
    Pageable pageable = new PageRequest(0,3, Sort.Direction.D
    Page<Customer> page = repository.findByName("bauer"
    //查询结果总行数
    System.out.println(page.getTotalElements());
    //按照当前分页大小，总页数
    System.out.println(page.getTotalPages());
    //按照当前页数、分页大小，查出的分页结果集合
    for (Customer customer: page.getContent()) {
        System.out.println(customer.toString());
    }
    System.out.println("-----")
}
```

从示例代码的注释当中可以看到Page对象的相关参数及值的说明，更详细的用法，参考PageRequest源码。

小结：怎么样，是不是很简单很方便？！

另：PageRequest.java源码在上一篇文章已经有了。

参考：

官方文档：<https://docs.spring.io/spring-data/jpa/docs/current/reference/html>

API官方文档：<http://docs.spring.io/spring-data/data-jpa/docs/current/api/>

JPQL文档：<http://www.blogjava.net/calmJava/archive/2011/04/01/347450.html>

DEMO示例：<https://github.com/icnws/spring-data-jpa-demo>

[Spring Data](#)

阅读	评论	喜欢
289	0	0

[喜欢](#)[SpringForAll](#)

文章	问答	粉丝
79	6	236

[关注](#)